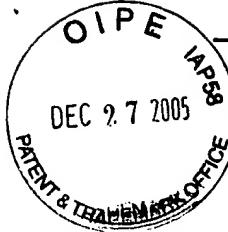


IN THE UNITED STATES PATENT AND TRADEMARK OFFICE



Application No.: 09/739,618  
Filed: December 18, 2000

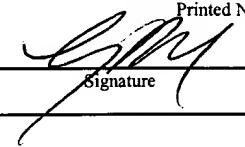
Inventor(s):  
John H. Howard

§ Examiner: Duong, Thomas  
§ Group/Art Unit: 2145  
§ Atty. Dkt. No: 5181-59100

§ I hereby certify that this correspondence is being deposited with  
§ the United States Postal Service with sufficient postage as first  
§ class mail in an envelope addressed to Commissioner for  
§ Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the  
§ date indicated below.

Lawrence J. Merkel

Printed Name



12/21/05

Signature Date

Title: Object-Based Storage  
Device with Improved  
Reliability and Fast Crash  
Recovery

**ATTN: Mailstop AF**  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

Dear Sir:

Applicant requests review of the final rejection in the above-identified application. No amendments are being filed with this request. This request is being filed with a notice of appeal. Independent claim 40 is rejected under 35 U.S.C. § 102(b) as being anticipated by Senator et al., U.S. Patent No. 5,761,677 ("Senator"). Independent claims 2, 8, 12, 22, and 29 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Senator in view of Zheng et al., U.S. Patent No. 6,571,259 ("Zheng"). Applicant sets forth the clear errors in the rejections below. Please note that for brevity, only the primary arguments directed mainly to the independent claims are presented, and that additional arguments, e.g., directed to the subject matter of the dependent claims, will be presented if and when the case proceeds to Appeal.

Independent Claims 2, 12, 22, and 29

Applicant respectfully submits that each of independent claims 2, 12, 22, and 29 recites a combination of features not taught or suggested in the cited art. For example, claim 2 recites a combination of features including: "a non-volatile memory storing a first inode locating a first file in said storage and also storing a journal comprising a list of committed inodes; and a block

manager ... configured to atomically update said first file in response to a commit of said first file by writing said second inode to said non-volatile memory ... and wherein said block manager is configured to record said second inode in said journal".

Applicant respectfully submits that Senator in view of Zheng fails to form a *prima facie* case of obviousness, because Senator teaches away from the alleged combination and because the combination fails to teach or suggest each and every feature of the claim.

Firstly, Senator teaches away from making the combination. The Office Action relies on Zheng to teach a journal. Senator also discloses, in the background, the existence of journals of changes to file system records (e.g. Senator, col. 1, lines 28-35). However, Senator teaches away from using such journals/data logs in his invention (on which the Office Action relies for other features of claim 2). For example, Senator teaches "The VERSION module 107 provides for various versions of a file without the need for data copy or log operations". (Senator, col. 3, lines 39-40). In fact, **avoiding the use of such journals/data logging is a central feature of** Senator's invention: "Various versions of a computer file are provided without requiring copying the file or logging changed data, so that the files have consistent user data" (Senator, abstract, lines 1-3); and "In accordance with the instant invention, the above problems of inconsistency and throughput bottlenecks have been solved by providing for various versions of a file without the need for data copy or log operations" (Senator, col. 1, lines 63-65). See also Senator, col. 2, lines 13-15; col 5 lines 43-47. In fact, Senator handles the consistency of file data and multiple versions of files in a completely different way, storing pointers to current and previous inodes in the inodes themselves (see, e.g., Senator, Figs 3c-3e, elements 305 and 307 and col. 5, lines 12-25). Even the section of Senator used in the Office Action as evidencing Senator's acknowledgement of journals/data logging teaches away from such a combination: "No separate data logs need be kept for purposes of data consistency" (Senator, col. 2, lines 26-27).

Secondly, the alleged combination fails to teach or suggest each and every feature of the claim. The Office Action relies on Zheng to teach a journal. While Zheng does teach a file system index 37, that index is "in-memory". Similarly, the Office Action refers to Zheng's LRU list 36, which is also in-memory. Zheng's teaching further include the following: "All disk-block reservation and pre-allocation mapping are in the memory, and after a crash, they are automatically discarded" (See Zheng, abstract, for example). Thus, Zheng intends for all in-memory information (including the file system index 37 and the LRU list 36) to be automatically

discarded in a crash. These teachings do not teach or suggest "a non-volatile memory storing a first inode locating a first file in said storage and also storing a journal comprising a list of committed inodes". Rather, **the described operation in Zheng is that of volatile memory**. The Response to Arguments in the present Office Action states that the reference does not use the term "volatile memory", and argues that recovering from a system failure by restoring the database to its consistent state existing just after commitment of the last complete transaction indicates that the memory must be non-volatile. Applicants respectfully disagree. **Zheng clearly describes the memory operating in a volatile fashion**. Furthermore, it would not be obvious to modify Zheng's teachings to store his file system index 37 or LRU list 36 in a non-volatile memory, as it would change the intended purpose of discarding all in-memory information. In summary, storing the file system index 37 and the LRU list 36 in a memory and automatically discarding such data, as taught in Zheng, does not teach or suggest "a non-volatile memory storing a first inode locating a first file in said storage and also storing a journal comprising a list of committed inodes". For at least the above stated reasons, Applicant submits that the rejection of claim 2 over the alleged combination of Senator and Zheng is not supported, and should be withdrawn. For similar reasons, Applicants submit that the rejection of claims 12, 22, and 29 are not supported.

#### Independent Claim 8

Claim 8 recites a combination of features including: "a storage coupled to receive said one or more write commands ... wherein said storage is configured to copy one or more blocks of said file to a copied one or more blocks, said one or more blocks updated by said one or more write commands, and wherein said storage is configured to update said copied one or more blocks with write data corresponding to said one or more write commands".

The current Office Action alleges that the combination of Senator and Zheng renders claim 8 obvious, but the rejection is grouped with that of claims 2, 12, 22 and 29 discussed above. The above highlighted features are not treated in that rejection. Accordingly, a **proper *prima facie* case of obviousness of claim 8 has not been established**, because EACH and EVERY feature in claim 8 has not been identified in the prior art.

In the Response to Arguments Section, the Office Action alleges that Senator teaches the above highlighted features at col. 2, lines 13-46 and col. 4, line 45-col. 5, line 48. Applicant respectfully disagrees, and asserts that these sections teach various details of making file updates

without copying data from old blocks to newly allocated blocks. For example, the following teachings are illustrative: " a module responsive to a system call argument to allocate another node in the file system tables and to copy the data block allocations from the old node into the newly allocated node. Both nodes now contain the same data block allocation information.

Shadow pointers are set in the old node to point to the new node and set in the new node to point to the old node. Changes to the actual data are now made with respect to the new node and fresh data blocks are allocated for the changed blocks." (Senator, col. 2, lines 15-23). Senator includes no teaching of copying data from a previous data block to the newly allocated data block when an update is made. In fact, Senator repeatedly teaches that no file data is copied in his system: "The VERSION module 107 provides for various versions of a file without the need for data copy or log operations". (Senator, col. 3, lines 39-40). Avoiding the use of such copying is a central feature of Senator's invention: "Various versions of a computer file are provided without requiring copying the file or logging changed data, so that the files have consistent user data" (Senator, abstract, lines 1-3); and "In accordance with the instant invention, the above problems of inconsistency and throughput bottlenecks have been solved by providing for various versions of a file without the need for data copy or log operations" (Senator, col. 1, lines 63-65). See also Senator, col. 2, lines 13-15; col 5 lines 43-47. While Senator copies allocation information from node to node, no copying of data from old blocks to newly allocated blocks is taught in Senator. For at least the above stated reasons, Applicant submits that the rejection of claim 8 is not supported by Senator and the rejection should be withdrawn.

#### Independent Claim 40

Claim 40 recites a combination of features including:

a computing node configured to generate a plurality of write commands to update a first file and a commit command defined to commit the updates to the first file;

an interconnect to which the computing node is coupled, wherein the computing node is configured to transmit the plurality of write commands and the commit command on the interconnect; and

a storage coupled to the interconnect and configured to store the first file, wherein the storage is configured to atomically update the first file to reflect the plurality of write commands responsive to the commit command.

Senator teaches a set of software modules (e.g. the reap module, rollback module, and version module) which execute on a host system. These modules communicate with the storage devices via the I/O system (for UFS) or the network I/O system (for NFS). That is, the modules are on the same side of the I/O systems as the user applications and other software that cause writes to files. Senator teaches: "The VERSION module 107 provides for various versions of a file without the need for data copy or log operations. Module 107 receives an input signal from either the COMMIT system call or the FSYNC call. The ROLLBACK module 109 restores a previous version of a file by associating the file name with the metadata for the previous version. The REAP module removes previous versions of a file that are determined to be no longer needed and impractical to retain indefinitely." Accordingly, these modules handle the creation and deletion of file versions. None of this teaches or suggests: "a computing node configured to generate a plurality of write commands ...an interconnect to which the computing node is coupled ...and a storage coupled to the interconnect and configured to store the first file, wherein the storage is configured to atomically update the first file to reflect the plurality of write commands responsive to the commit command." For at least the above stated reasons, Applicant submits that the rejection of claim 40 is not supported by Senator and the rejection should be withdrawn.

For at least the above stated reasons, Applicant submits that the application is in condition for allowance, and a Notice thereof is respectfully requested. If any extensions of time (under 37 C.F.R. § 1.136) are necessary to prevent the above referenced application(s) from becoming abandoned, Applicant(s) hereby petition for such extensions. If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5181-59100/LJM. Also enclosed herewith are the following items:

- Return Receipt Postcard
- Notice of Appeal

Respectfully submitted,



\_\_\_\_\_  
Lawrence J. Merkel  
Reg. No. 41,191  
AGENT FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.  
P.O. Box 398  
Austin, TX 78767-0398  
Phone: (512) 853-8800